Iterative research method applied to the design and evaluation of a dynamic multicast routing scheme

Dimitri Papadimitriou[1], Florin Coras[2], Alberto Rodriguez[2], Valentin Carela[2], Davide Careglio[2], Lluís Fàbrega[3], Pere Vilà[3], Piet Demeester[4]

Alcatel-Lucent Bell Labs, Antwerp, Belgium
dimitri.papadimitriou@alcatel-lucent.com
Universitat Politècnica de Catalunya, Barcelona, Spain
{fcoras,arnatal,vcarela,careglio}@ac.upc.edu
Universitat de Girona, Girona, Spain
{lluis.fabrega,pere.vila}@udg.edu
Ghent University and iMinds, Ghent, Belgium
piet.demeester@intec.ugent.be

Abstract. Following the iterative research cycle process, this chapter elaborates a methodology and documents the steps followed for the design of a dynamic multicast routing algorithm, referred to as Greedy Compact Multicast Routing. Starting from the design of the dynamic multicast routing algorithm, we then evaluateby simulation on large-scale topologies its performance and compare them with the Abraham compact multicast routing scheme and two other reference schemes, namely the Shortest Path Tree (SPT) and the Steiner Tree (ST) algorithm. Performance evaluation and comparison include i) the stretch of the multicast routing paths also referred to as multicast distribution tree (MDT), ii) the memory space required to store the resulting routing table entries, and iii) the total communication or messaging cost, i.e., the number of messages exchanged to build the MDT. However, such performance evaluation is a necessary but not a sufficient condition to meet in order to expect deployment of multicast routing. Indeed, if one can determine that traffic exchanges are spatially and temporally concentrated, this would provide elements indicating the relevance for the introduction of such scheme in the Internet. Otherwise (if traffic exchanges are spatially and temporally diverse, i.e., highly distributed), then very few of them would benefit from a (shared) point-to-multipoint routing paths and multicast routing scheme would be less useful. For this purpose, we have conducted a multicast tree inference study. In turn, data and results obtained from these studies providesmore realistic scenarios for emulation experiments against the currently deployed approach combining MBGP and PIMdeployed in IPTV or mVPN context.

Keywords: multicast routing, compact, experimental, performance, evaluation

1 Introduction

The Future Internet Research and Experimentation (FIRE) initiative aims to realize a "research environment for investigating and experimentally validating highly innovative and revolutionary ideas" towards new paradigms for theInternet by bridging multi-disciplinary long-term research and experimentally-driven large-scale validation. FIRE foundational objectives were:

- Creation of a multi-disciplinary, long term research environment for investigating and experimentally validating highly innovative and revolutionary ideas for new networking architectures and service paradigms;
- Promotion of experimentally-driven yet long-term research, joining the two ends of academy-driven visionary research and industry-driven testing and experimentation, in a truly multi-disciplinary and innovative approach;
- Realization of a large scale European experimental facility, by gradually inter-connecting and federating existing and new "resource clusters" for emerging or future internet architectures and technologies.

These objectives further evolved toward the inception of experimentally-driven research as a visionary multi-disciplinary investigation activity, defining the challenges for and taking advantage of experimental facilities. Such investigation activity would be realized by means of iterative cycles of research, oriented towards the design and large-scale experimentation of new and innovative paradigms for the Internet - modeled as a complex distributed system. The refinement of the research directions should be strongly influenced by the data and observations obtained from experiments performed at previous iterations thus, being "measurement-based" which in turn requires the specification of the relevant criteria and metrics as well as their corresponding measurement tools. The rationale was thus clear: create a dynamic between elaboration, realization, and validation by means of *iterative cycles* of experimentation.

With the increasingof multimedia streaming/content traffic, multicast distribution process from a source to a set of destination nodes is (re-)gaining interest as a bandwidth saving technique competing with or complementing cached content distribution. Nevertheless, the scaling problems faced in the 90's when multicast routing received main attention from the research community remain mostly unaddressed since so far. Indeed, routing protocol dependent multicast routing schemes such as Distance Vector Multicast Routing Protocol (DVMRP) and Multicast Open Shortest Path First (MOSPF) have been replaced by routing protocol independent routing schemes such as Protocol Independent Multicast Sparse Mode (PIM-SM) [1] and Core Base Trees (CBT) [2]. During last decade, the Single Source Multicast (SSM) variant of PIM, referred to as PIM-SSM [3], has been deployed in the context of IPTV within Internet Service Provider's network (intra-domain multicast). However inter-domain multicast has failed to be widely adopted by most ISPs. The reasons, among others, result fromtherelative complexity of the protocol

architecture. Overlaying multicast routing on top unicast routing topology suffers from the same scaling limits as unicast (shortest-path) routing with the addition of the level of indirection added by the multicast routing, the limits of Multicast Source Discovery Protocol (MSDP) which prevents shared trees between domains (thus, it defeats the objectives of PIM-SM) and its address space structure (multicast addressing also requires firewall upgrades to recognize Class-D addresses). On the other hand, deploying multicast routing requires routing equipment upgrade (both hardware and software) whereas the corresponding cost cannot be compensated by multicast service revenues when the ISP doesn't itself provide access to multicast receivers (or sources).Further analysis on deployment Issues for the IP multicast routing and architecture can be found in [4].

As part of the work conducted in the EULER FP7 project [5], we started bydesigning a dynamic multicast routing algorithm, referred to as Greedy Compact Multicast Routing (GCMR) [6]. This leaf-initiated routing schemewhich runs independently of the unicast routing scheme (and does not share any routing state information) is specialized for the construction of multicast routing paths (or multicast distribution trees) from any source to any set of destination nodes (or leaf nodes). We have then evaluated the performance of the proposed GCMR schemeand compare them for the same topologies with the Abraham compact multicast routing scheme [7] and two other reference schemes, the Shortest Path Tree (SPT) and the Steiner Tree (ST) algorithm. The performance evaluation and comparison by simulation of these multicast routing algorithms include: i) the stretch of the multicast routing paths it produces, ii) the memory space required to store the resulting routing table entries, and iii) the total communication or messaging cost, i.e., the number of messages exchanged to build the entire Multicast Distribution Tree (MDT).

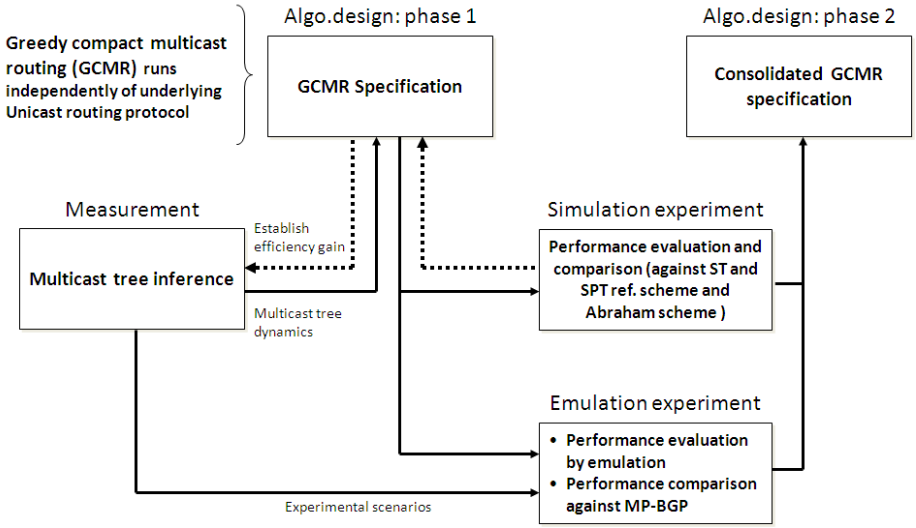


Figure 1. Experimental methodology

However, such performance evaluation is a necessary but not a sufficient condition to meet in order to expect deployment of multicast routing. Indeed,by determining that traffic exchanges are spatially and temporally concentrated, one would increasethe relevance for the introduction of such scheme in the Internet.Otherwise, if traffic exchanges are spatially and temporally diverse (highly distributed), then very few of them would benefit from a (shared) multicast routing paths and thus multicast routing scheme would be less useful. For this purpose, we have conducted a multicast tree inference study. In turn, the data and results obtained from these studies enable to design more realistic scenarios for the emulation experiments and the performance comparison against the currently deployed approach combining Multiprotocol BGP specified in RFC 4760 [8] also referred to as MBGP or MP-BGP (for discovery purposes) and PIM (for signaling purposes) deployed in context of IPTV or multicast VPN (mVPN). Figure 1 summarizes our methodology together with the various iterative research cycles until the step where a consolidated version of the GCMR scheme combining all experimental results can be specified.

2 Dynamic multicast routing

Dynamic multicast routing algorithms enable the construction of point-to-multipoint routing paths from any source to any set of destination nodes (or leaf nodes). The tree determined by a multicast routing path is commonly referred to as a Multicast Distribution Tree (MDT) as it enables the distribution of multicast traffic from any source to any set of leaf nodes. By means of such dynamic routing scheme, MDTs can dynamically evolve according to the arrival of leaf-initiated join/leave requests. The multicast routing algorithm creates and maintains the set of local routing states at each node part of the MDT. From this state, each nodes part of the MDT can derive the required entries to forward the multicast traffic received from a given source to its leaves.

2.1 Greedy Compact Multicast Routing (GCMR)

In [6],we introduce a dynamic compact multicast routing algorithm that enables the construction of multicast distribution trees (also referred to as multicast routing paths) for the distribution of multicast traffic from any source to any set of leaf nodes. The objective of the proposed GCMR scheme is to minimize the routing table sizes of each node part of the MDT at the expense of i) routing the packets on multicast routing paths with relative small deviation compared to the optimal stretch obtained by the Steiner Tree (ST) algorithm, and ii) higher communication cost compared to the Shortest Path Tree (SPT) algorithm. For this purpose, the GCMR algorithm reduces the local storage of routing information by keeping only direct neighbor-related entries rather than tree structures (as in ST) or network graph entries (as in both SPT and ST). The proposed algorithm relies on the information obtained locally and proportionally to the node degree instead of requiring knowledge of the global

topology information (proportional to the network size)while still providing the least cost next hop during the MDT construction. In other terms, the GCMR algorithm requires only the maintenance of local topology information and does not rely on the knowledge of the global topology information or involve the construction of global network structures such as sparse covers. The information needed to reach a given multicast source is acquired by means of a two-stage search process that returns the upstream node along the least cost branching path to the MDT sourced at s. This process is triggered whenever a node decides to join a given multicast source s, root of the MDT. After a node becomes member of an MDT, a multicast routing entry is dynamically created and stored in the local Tree Information Base (TIB). From these routing table entries, multicast forwarding entries are locally instantiated.

More formally, consider a network topology modeled by an undirected graph G = (V,E,c), where the set V, |V| = n, represents the finite set of nodes or vertices (all being multicast capable), the set E, |E| = m, represents the finite set of links or edges, and c a non-negative cost function c: $E \rightarrow Z+$ that associates a non-negative cost c to each link $(u,v) \in E$. For $u, v \in V$, let c(u,v) denote the cost of the path p(u,v) from u to v in G, where the cost of a path is defined as the sum of the costs along its edges. Let S, $S \subset V$, be the finite set of source nodes, and $s \in S$ denote a source node. Let D, $D \subseteq V\backslash\{S\}$, be the finite set of all possible destination nodes that can join a multicast source s, and $d \in D$ denote a destination (or leaf) node. A *multicast distribution tree* $T_{s,M}$ is defined as an acyclic connected sub-graph of G, i.e., a tree rooted at source $s \in S$ with leaf node set M, $M \subseteq D$. During the MDT construction, the routing information needed to reach a given multicast distribution tree is acquired by means of an incremental two-stage search process. This process, triggered whenever a node decides to join a given multicast source, starts with a local search covering the leaf node's neighborhood. If unsuccessful, the search is performed over the remaining unexplored topology (without requiring global knowledge of the current MDT). The returned information provides the upstream neighbor node along the least cost branching path to the MDT rooted at the selected multicast source node. The challenge consists thus in limiting the communication cost, i.e., the number of messages exchanged during the search phase, while keeping an optimal stretch - memory space tradeoff.

As stated before, the reduction in memory space consumed by the routing table entries results however in higher communication cost compared to the reference algorithms, namely the SPT and the ST. Higher cost may hinder the applicability of our algorithm to large-scale topologies such as the Internet. Hence, to keep the communication cost as low as possible, the algorithm's search process is segmented into two different stages. The rationale is to put tighter limits on the node space by searching locally in the neighborhood (or vicinity) of the joining leaf node before searching globally. Indeed, the likelihood of finding a node of the MDT within a few hops distance from the joining leaf is high in large topologies (whose diameter is logarithmically proportional to its number of nodes) and this likelihood increases with the size of the MDT. Hence, we segment the search process by executing first a local

search covering the leaf node's vicinity ball, and, if unsuccessful, by performing a global search over the remaining topology. By limiting the size (or order) of the vicinity ball while taking into account the degree of the nodes it comprises, one ensures an optimal communication cost. For this purpose, a variable path budget π_β is used to limit the distance travelled by leaf initiated requests to prevent costly (in terms of communication) local search or global search. Additionally, as the most costly searches are resulting from the initial set of leaf nodes joining the multicast traffic source, each source constructs a domain (referred to as source ball). When a request reaches the boundary of that domain it is directly routed to the source.

2.2 Comparison with existing IP multicast routing

Independence from the underlying unicast routing algorithm is the fundamental concept underlying multicast routing schemes such as Protocol Independent Multicast (PIM). Its variants for any-source multicast (PIM-SM) [1] and single-source multicast (PIM-SSM) [3] are the most commonly deployed routing protocols even if limited in scope to single carrier networks. It is important though to distinguish between algorithmic independence, i.e., no computational coupling from informational independence, i.e., PIM makes use of the unicast routing table to enable control message exchanges (join, prune, etc.). Indeed, overlaying multicast routing on top of unicast suffers however from the same scaling limitations as current unicast routing with the addition of the level of indirection added by the multicast routing application. Multicast routing protocol enables routers to build a (logical) delivery tree between the sender(s) and receivers of a multicast group. Multicast routing table includes the Multicast Routing Information Base (MRIB) and the Tree Information Base (TIB). The MRIB is the topology table, typically derived from the unicast routing table, which carries multicast-specific topology information. The TIB is the collection of routing state created from the exchange of join/prune messages. This table stores the state of all multicast distribution trees at that node. The implication being that in case of topology change, unicast routing states have to re-converge to a new stable state before multicast routing states can themselves re-converge.

Moreover, we also observe that the scaling problems experienced by these routing protocols and more generally all multicast routing approaches developed by the research community, remain largely unaddressed since so far. Indeed, multicast currently operates as an addressable IP overlay (Class D group addresses) on top of unicast routing topology, leaving up to an order of 100millions of multicast routing table entries. Hence, the need to enable multicast routing paths (for bandwidth saving purposes) while keeping multicast addressing at the edges of the network and building shared but selective trees inside the network. When used in combination to GCMR, multicast forwarding relies on local port information only. Thus, the memory capacity savings comes from i) keeping 1:N relationship between network edge node and the number of multicast groups (N), and ii) local port-based addressing for the local processing of multicast traffic. Further, we argue that the GCMR scheme, by providing the best memory-space vs. stretch tradeoff, can possibly address the

memory scaling challenges without requiring the deployment of an underlyingunicast routing scheme.

The version 4 of the Border Gateway Protocol (BGPv4) has also been extended to support multicast discovery protocol. This extension relies on the multiprotocol BGP (MBGP) feature defined in RFC 4760 [8]. The multi-protocol capability of BGP enables multicast routing and the connectionof multicast topologies within and between BGP autonomous systems. In other words, multiprotocol BGP (MBGP) is an enhanced BGP that carries IP multicast routes. BGP carries two sets of routes, one set for unicast routing and one set for multicast routing. The routes associated with multicast routing are used by the Protocol Independent Multicast (PIM) to build data distribution trees. More recently, this feature has been further extended in RFC 6513 [9] and BGPv4 can now also be used as multicast signaling protocol; hence, avoiding the use of PIM.

2.3 Comparison with other compact multicast routing

As far as our knowledge goes, prior work on compact multicast routing is, mainly concentrated around the routing schemes developed in the seminal paper of Abraham [7]. One of the reasons we can advocate is that despite the amount of research work dedicated to compact unicast routing, current schemes are not yet able to efficiently cope with the dynamics of large scale networks which is the prime characteristic of dynamic multicast routing schemes.

More formally, the Abraham scheme relies on the off-line construction of a bundle $\mathcal{B}_k = \{TC_{k,2i}(G) \mid i \in I\}$ of sparse tree covers of the graph G, $TC_{k,2i}(G)$, where $k = \log(n)$. Sparse covers are grown from a set of center nodes $c(T_i(v))$ located at distance at most $k.2^i$ from node $v \in V$.By $T_i(v)$, we denote the tree in the sparse tree cover $TC_{k,2i}(G)$ that contains the ball $B(v,2^i)$. For each $i \in I$ and $T \in TC_{k,2i}(G)$, the center node $c(T(v))$ stores the labels ofall nodes[1]contained in the ball $B(v,2^i)$, i.e., the ball centered on node v of radius 2^i. Further, the SPlabel(v) stores the label $\lambda(T,c(T))$ for each $T \in \mathcal{B}(v)$, defined as set of all tree covers T in the bundle $\mathcal{B}_k$ such that $v \in T$. In addition, each node $v \in V$ stores the tree routing information $\mu(T,v)$ for all the trees in its own label SPlabel(v). When a leaf node u joins an MDT, it first determines whether or not one of the MDT nodes is already included in its local tree routing information table. If this is the case, node u sends the join request to the center node $c(T_i(v))$ of the tree $T_i(v)$ that is covered by the MDT at node v. The center node $c(T_i(v))$ then passes the label $\mu(T_i(v),u)$ so that the selected MDT node v can forward the multicast traffic to the newly joining leaf node u. Otherwise, the leaf node u has to obtain via its center node the set of MDT nodes the tree currently includes. Among all index $i \in I$, node u then selects the tree $T_{i*}(v)$, $v \in$ MDT, whose intersection with its bundle $\mathcal{B}(u)$ is minimum. Once the node, say v, part of this intersection is selected ($T_{i*}(v) \in \mathcal{B}(u)$),

[1]For simplicity, we present here the label-dependent variant of the scheme. In the name-independence version, center nodes store label mappings from names to nodes.

leaf node u directs the join request to the associated center node $c(T_{i*}(v))$. The latter passes a label $\mu(T_{i*}(v),u)$ so that the selected MDT node v can forward the incoming multicast traffic to the newly joining leaf node u. In both cases, in order for the source node s to reach node u, node v has to propagate the tuple $[v,c(T_{i*}(v)),\mu(T_{i*}(v),u)]$ to source s. Finally, the leaf node u updatesall nodes covered by its balls $B(u,2^i)$ to allow them joining the MDT at node u.

Compared to the Abraham compact multicast routing scheme [7], the GCMR name-independent compact multicast routing algorithm is i) leaf-initiated since join requests are initiated by the leaf nodes; however, contrary to the Abraham scheme it operates without requiring prior local dissemination of the node set already part of the MDT or keeping specialized nodes informed about nodes that have joined the MDT, and ii) dynamic since requests are processed on-line as they arrive without re-computing and/or re-building the MDT. Moreover, our proposed algorithm is iii) distributed since transit nodes process homogeneously the incoming requests to derive the least cost branching path to the MDT without requiring any centralized processing by the root of the MDT or any specialized processing by means of pre-determined center nodes, and iv) independent of any underlying sparse cover construction grown from a set of center nodes (which induce node specialization driving the routing functionality): the local knowledge of the cost to direct neighbor nodes is sufficient for the proposed algorithm to properly operate. It is important to emphasize that the sparse cover underlying the execution of the Abraham scheme is constructed off-line and requires global knowledge of the network topology to properly operate.Moreover, this routing schemeis oblivious, i.e., the path from the source to a given leaf is irrespective of the current set of leaves (even if its iterative construction implies the "local" dissemination of information related to nodes that have already joined the tree) whereas the GCMR scheme is adaptive. The resulting adaptation cost remains to be characterized.

3 Performance analysis

The performance of the GCMR algorithm, further documented in [6] are evaluated in terms of the stretch of the multicast routing paths it produces, the size and the number of routing table entries, and the communication/messaging cost. Performances are evaluated by simulation on synthetic power-law graphs (generated by means of the Generalized Linear Preference (GLP) model [10]) and the CAIDA map of the Internet topology bothcomprising 32k nodes. Performance results are compared to the multicast routing algorithms (the Shortest Path Tree and the Steiner Tree algorithm) and the Abraham scheme.

3.1 Reference routing schemes: ST and SPT

The execution scenario considers the construction of point-to-multipoint routing paths for leaf set of increasing size from 500 to 4000 nodes (selected randomly) with

increment of 500 nodes. Each execution is performed 10 times by considering 10 different multicast sources. We compare the performance of the GCMR algorithm to the Shortest-Path Tree (SPT) which provides the reference for the communication cost and the Steiner Tree (ST) algorithms which provides the reference in terms of stretch. In order to obtain the near optimal solution for the ST, we consider a ST-Integer Linear Programming formulation adapted from [11] for bi-directional graphs.

The communication cost for the ST measures at each step of the MDT construction the number of messages initiated by nodes part of the MDT. Hence, although the ST is computed centrally, the communication cost accounts for the total number of messages exchanged during the MDT building process as a dynamic scenario would perform.

Multiplicative stretch:defined as the cost ratio between the point-to-multipoint routing paths (underlying the MDT) produced by the proposed scheme and the minimum Steiner Tree. We also compare the cost ratio between the point-to-multipoint routing path produced by the SPT and the minimum Steiner Tree (ST).

- *GLP Topology*: as shown in Figure 2a, the multiplicative stretch for the proposed algorithm is slightly higher than 1 for the GLP topology. As the leaf set increases from 500 to 4000 nodes, its trend curve decreases from 1.09 (maximum value reached for 500 leaf nodes) to 1.05 (minimum value reached for 4000 leaf nodes). Compared to the SPT stretch, our algorithm maintains an average gain of 4% along the different group sizes.
- *CAIDA Map*: as shown in Figure 2b, the multiplicative stretch for the proposed algorithm is slightly higher than 1 for the CAIDA topology. As the leaf set increases from 500 to 4000 nodes, its trend curve decreases from 1.08(maximum value reached for 500 leaf nodes) to 1.03 (minimum value reached for 4000 leaf nodes). Compared to the SPT stretch, our algorithm maintains a maximum deterioration of 4% for sets of 500 leaf nodes; this deterioration becomes negligible as the size of the leaf node sets increases.

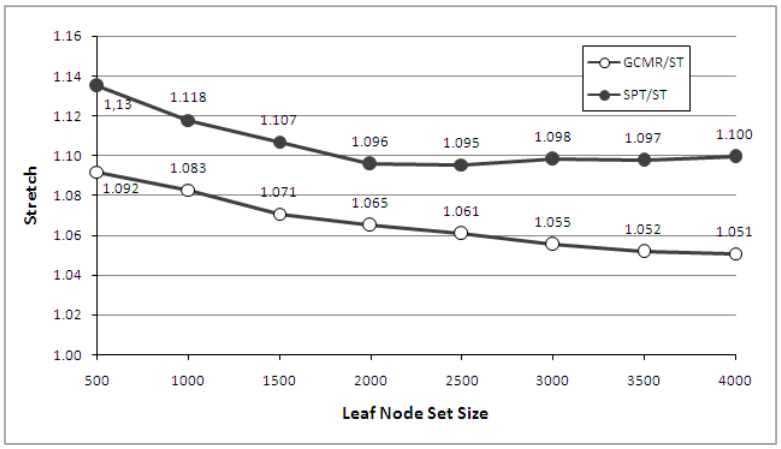


Figure 2a. Stretch as a function of Leaf Node Set Size

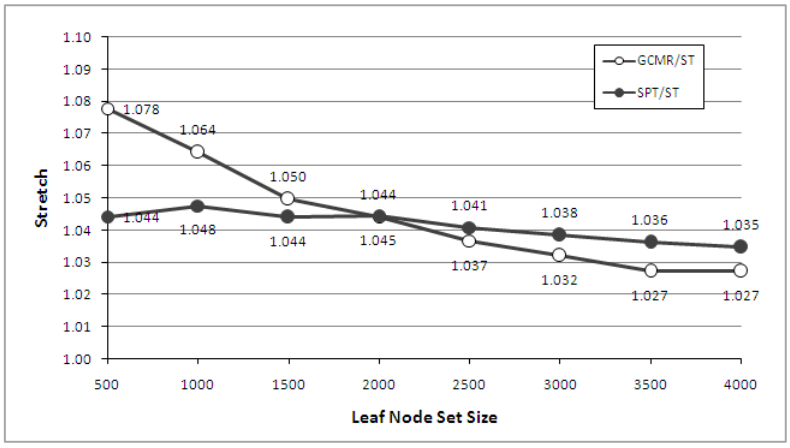


Figure 2b. Stretch as a function of Leaf Node Set Size

Storage: is concerned with the memory space required to store the routing tables entries (underlying the MDT) and the relative gain we obtain in terms of the ratio between the total number of RT entries produced by our algorithmand the total number of RT produced by the reference algorithms. This ratio provides an indication of the achievable reduction in terms of the memory capacity required to store the routing table entries produced by these algorithms.

- *GLP Topology*: the GCMR scheme produces significantly less RT entries that the ST and SPT reference algorithms. The highest number of RT entries is obtained for a set of 4000 leaf nodes: 10154 RT entries. This value is 4,10 times smaller than the number of RT entries produced by the ST algorithm (41643 RT entries) and 27,92 times smaller than the number of the RT entries produced by the SPT algorithm (283477 RT entries). Figure 3a illustrates the relative gain expressed in terms of the ratio between the total number of RT entries produced by the ST and

the SPT references and our algorithm. An increasing gain can be observedas the size of theleaf node set decreases from 4,10 (leaf set of 4000 nodes) to 20,34 (leaf set of 500 nodes) compared to the ST algorithm and from 27,92 (leaf set of 4000 nodes) to 166,08 (leaf set of 500 nodes) compared to the SPT algorithm.

- *CAIDA Map*: the GCMR scheme producessignificantly less routing table entries that the ST and SPT reference algorithms. The highest number is obtained for leaf sets of 4000 nodes: 13169 RT entries. This value is 3,21 times smaller than the number of RT entries produced by the ST algorithm (42277 RT entries) and 14,38 times smaller than the number of RT entries produced by the SPT algorithm (189431 RT entries). Figure 3b illustrates the relative gain in terms of the ratio between the total number of RT entries produced by the ST and SPT references and GCMR. An increasing gain can be observedas the size of theleaf set decreases from 3,21 (leaf set of 4000 nodes) to 6,93 (leaf set of 500 nodes) compared to the ST algorithm and from 14,38 (leaf set of 4000 nodes) to 36,79 (leaf set of 500 nodes) compared to the SPT algorithm. Interestingly, the obtained gain values for the CAIDA map are smaller than those obtained for the GLP topology. This difference can be explained resulting from the difference in tree-depth: 6 (leaf set of 500 nodes) to 9 (leaf set of 4000 nodes) for the CAIDA map vs. 8 (leaf set of 500 nodes) to 11 (leaf set of 4000 nodes) for the GLP topology.

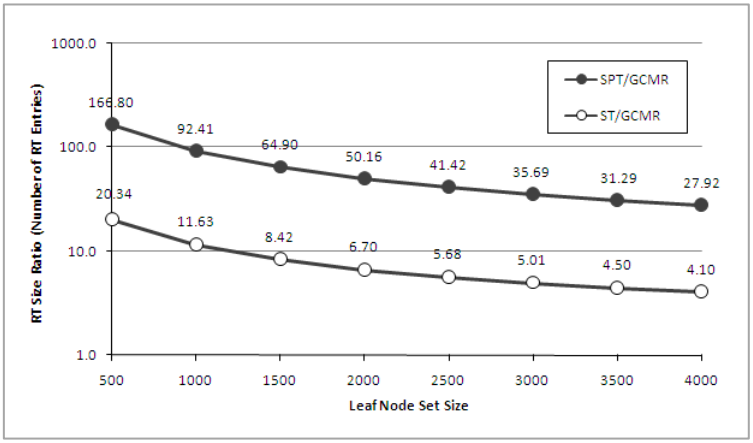


Figure 3a. RT Size Ratio as a function of Leaf Node Set Size

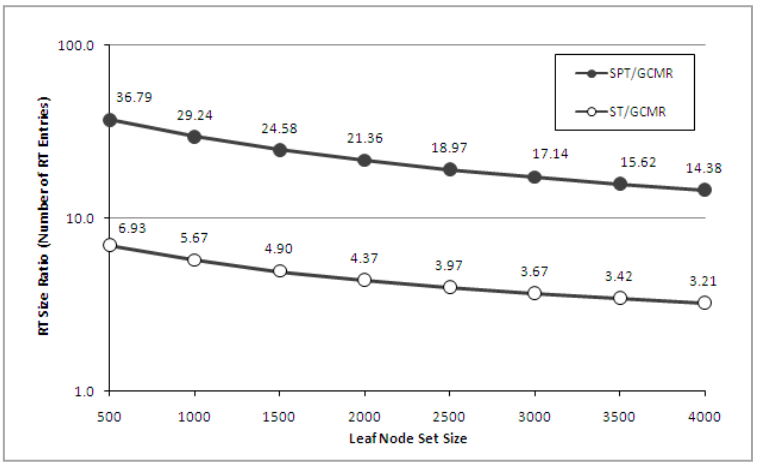


Figure 3b. RT Size Ratio as a function of Leaf Node Set Size

Communication Cost: defined as number of messages exchanged during the discovery search phase

- *GLP Topology*: as depicted in Figure4a, the communication cost ratio for the proposed algorithm is relatively high compared to the SPT even if much lower than the communication cost implied by the ST (not represented in this figure). Indeed, the communication cost ratio increases from 2,69 (leaf set of 500 nodes) to 8,17 (leaf set of 4000 nodes). This observation can be explained by the presence of high degree nodes (nodes that have a degree of the order to 100 or even higher) in power law graphs. However, as computed this communication cost does not take into account the evolution of the routing topology. This evolution impacts multicast routing algorithms such as the SPT that are strongly dependent on non-local unicast routing information compared to the proposed algorithm.Moreover, as shown in Figure 5, the communication cost of the proposed algorithm compared to the SPTcommunication cost, decreases as the number of nodes composing the leaf node set increases. This trend leads us to expect that a saturation level can be reached around a communication cost ratio not higher than 10 to 15 as the size of the lead node set continues to grow. It is worth mentioning that the memory and the capacity required to process communication messages are relatively limited.
- *CAIDA Map*: the same trend can be observed for the CAIDA Map (see Figure 4b) where the communication cost ratio between our scheme and the SPT algorithm increases from 7,88 (leaf set of 500 nodes) to 13,77 (leaf set of 4000 nodes). The difference observed between the CAIDA map and the GLP topology can be explained from the following observation the tree-depth differs by a unit (3 vs. 4). This difference induces a relatively higher cost of the SPT when running over the GLP topology.

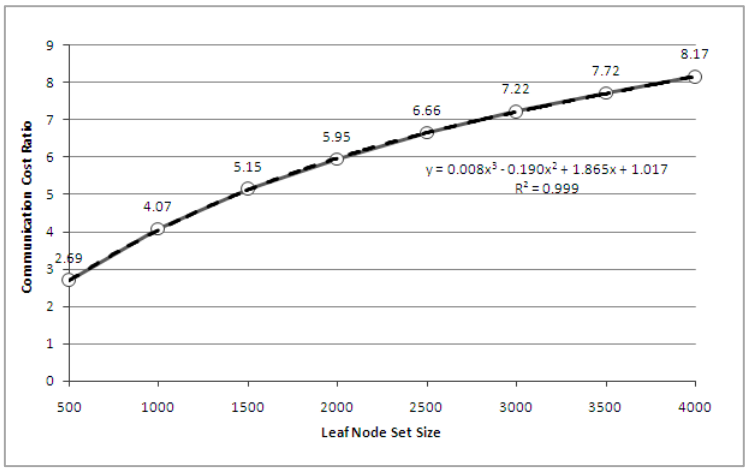


Figure 4a. Communication Cost Ratio as a function of Leaf Node Set Size

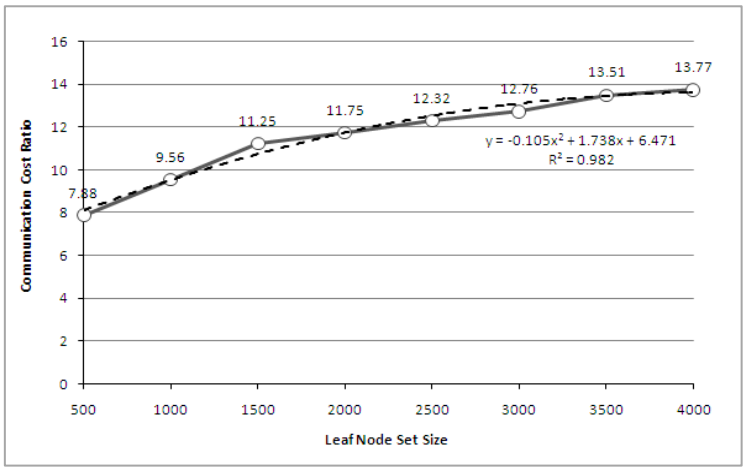


Figure 4b. Communication Cost Ratio as a function of Leaf Node Set Size

3.2 Abraham scheme

We further compare the performance in terms of the stretch of the point-to-multipoint routing paths and the memory space required by the GCMR algorithm and by theAbraham routing scheme for dynamic join only events.

Stretch: for the scheme allowing only dynamic join events, the MDT cost is given by Lemma7 of [7]. The authors determine that the proposed dynamic multicast algorithm is O(min{log n, log Δ}.log n) competitive compared to the cost of the optimal algorithm – Steiner Tree. In this formula, the factor Δ is the aspect ratio defined as the ratio between the maximum and the minimum distance d(u,v), for any

node pair u,v∈V. Considering an aspect ratio Δ of 6 and a network of 32k nodes the stretch upper bound is about 3.5. Thus the stretch upper bound of the point-to-multipoint routing path produced by the Abraham scheme, even if universal (applicable to any graph), is in the best casemore than 3 times higher than the one produced by our scheme. Simulation result show (see Figure 5) that this upper bound is not reached though the GCMR stretch is still twice better than the one obtained with the Abraham scheme. Note also that the comparative gain is weakly influenced by the value k (which determines the sparse cover construction, the higher the value the less the number of trees in the tree cover).

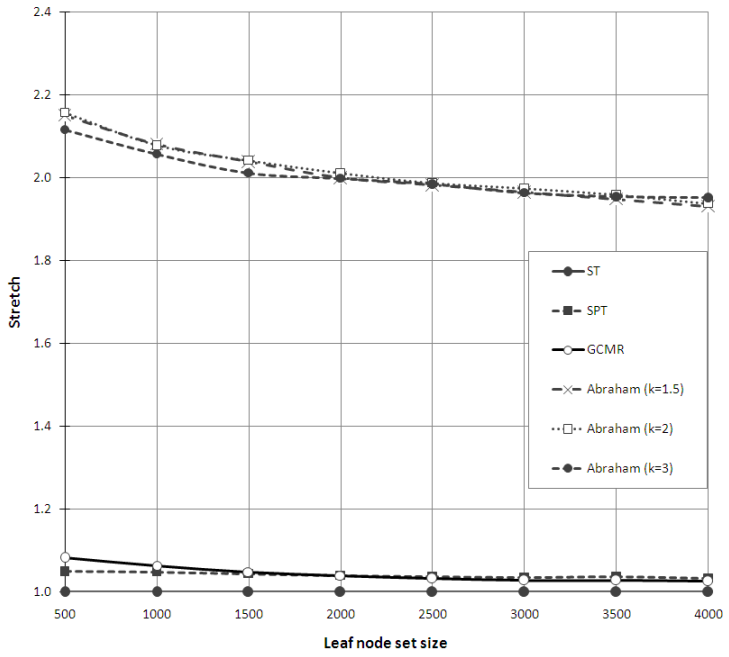


Figure 5. Stretch as a function of Leaf Node Set Size

Storage: the memory/storage requirement of the Abraham scheme includes i) the tree routing information μ(T,v) stored by each node v, for all trees in its own SPLabel(v) leading to a total storage of $O(\log^3 n.\log\Delta/\log\log n)$ bits, ii) for each i∈I and T∈$TC_{k,2i}(G)$, the center node c(T(v)) of each node v ∈ T that stores the labels ofall nodescontained in the ball $B(v,2^i)$ leading to a total storage over all radii of $O(kn^{1+1/k}\log\Delta)$ bits; in addition, each node v stores O(log Δ) labels of size $\tilde{O}(kn^{1/k})$ each leading to a total memory consumption of $\tilde{O}(kn^{1+1/k})$ bits.

To simplify comparison the GCMR memory space complexity is proportional to the number of tree nodes (2τ) and the Abraham scheme proportional to the number of nodes in network times the construction parameter k (k.n). This yields for k = 1 a ratio of 32 (4) for a leaf set of 500 (4000) nodes, for k = 2 a ratio of 64 (8) for a leaf set of

500 (4000) nodes, and for k = 3, a ratio of 96 (16) for a leaf set of 500 (4000). These numbers and trends remain to be further confirmed and validated.

4 Effective gain analysis

In this section, we aim at determining the rationale for introducing a multicast routing scheme as part of the Internet-wide routing system.This rationale is based on the following premises if traffic exchanges are spatially and temporally diverse, very few of them would benefit from a (shared) point-to-multipoint routing paths and thus multicast routing scheme would be useless; otherwise (if traffic exchanges are spatially and temporally concentrated), this would indicate the relevance for the introduction of such scheme in the Internet.

In the following we present the method adopted to classify applications from data traffic captures. We further analyze the benefits that a potential multicast routing scheme can provide. Finally, we analyze the traffic statistical characteristics of a streaming video application when transmitting a popular sport event.

4.1 Traffic classification

Traffic classification is a difficult problem that requires the use of very complex identification techniques due to the variable nature of Internet traffic and applications. Traditionally, the port numbers were used to classify the network traffic (e.g., well-known ports registered by the IANA). Nevertheless, nowadays it is widely accepted that this method is no longer valid due to its inaccuracy and incompleteness of its classification results. The first alternatives to the well-known ports method relied on the inspection of the packet payloads in order to classify the network traffic [12] [13] [14]. These methodsconsist of looking for characteristic signatures (or patterns) in the packet payloads. Although this solution could potentially be very accurate, its high resource requirements and limitations with encrypted traffic make its use unfeasible in current high-speed networks.

Instead, we developed therefore a traffic classification tool using NetFlow[15] data instead of packet-level traces. We use the well-known C4.5 decision tree technique [16] in order to analyze the impact of traffic sampling on the classification accuracy with Sampled NetFlow [17] [18]. To reduce the impact of traffic sampling on the classification accuracy, the tool implements an automatic Machine Learning (ML) algorithm that does not rely on any human intervention. More details can be found in [12].This tool has been applied to the traffic captured in the connection between the Anella Científica(Catalan NREN [13]) to RedIRIS (Spanish NREN [14]) and to the global Internet. The point of measurement is a 10Gbit Ethernet bi-directional link, which provides Internet connection to 60 different public entities and 50,000 users.

Figure 6 shows the obtained results where the outer ring illustrates the percentage of the outgoing traffic while the inner ring the incoming traffic with respect to the Catalan NREN. In Figure 6a, we can observe that the majority of the outgoing traffic belongs to theWeb/HTTP and the P2P application. For the incoming traffic the situation is different: P2P application is the third in terms of traffic percentage while multimedia applications become second. It is worth mentioning that in this figure, the term multimedia application refers to all video/audio applications that are not chat, Web/HTTP, P2P or games. In Figure 6b, we further classify some of the multimedia applications. In this figure, we include all video traffic obtained through web (like YouTube or DailyMotion), P2P(like SOPcast or pplive), or streaming (like Windows Media or Quicktime).

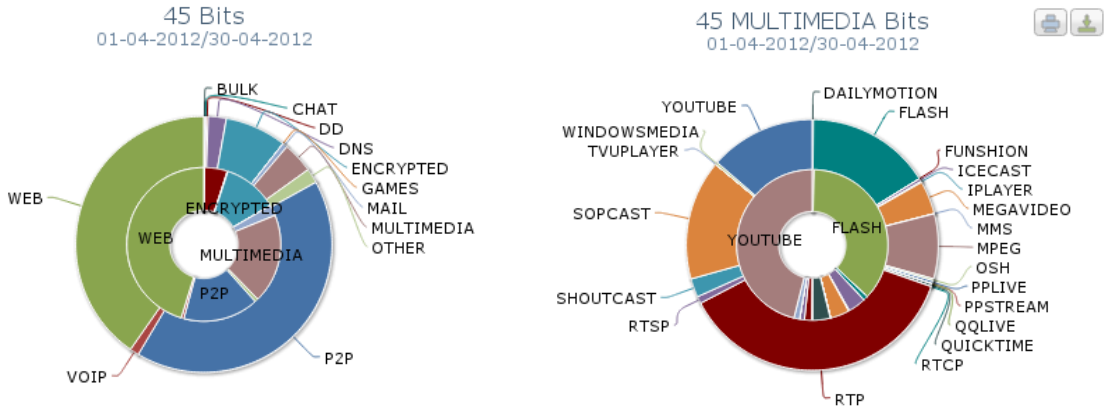


Figure 6. Outer ring is the outgoing traffic while the inner ring is the incoming traffic
a) Classification of the applications (left), and b) Classification of the multimedia traffic (right)

According to these results, the majority of traffic belongs to two types of application, namely Web/HTTP and P2P video streaming. For these very reasons, in the following sections we further analyze these applications to detect the conditions for which a multicast routing scheme would be beneficial.

4.2 Web/HTTP traffic analysis

This study aims at determining if Web/HTTPtraffic could be served in a one-to-many (multicast) to the requesting clients. For this purpose, we define a time frame *t*. We assume that such *t* is the time a web server can wait before transmitting the packets to the client. If more than one client requests the same content during such *t*, the server will only transmit once while the multicast routing is supposed to take care of replicating the traffic in the network. Given these assumptions, we make use of the same point of measurement described below (i.e., 10Gbit Ethernet link between Catalan NREN and Spanish NREN). We analyzeall traffic crossing the point of measurement during periods of 30 minutes and count all identical web content that has been transmitted during the same time frame *t*. If we define as c_i the number of

times the same content i has been transmitted, we define as traffic saving the percentage $\sum_i(c_i\text{-}1)/\sum_i c_i$. That is the content i would be transmitted only once if the multicast routing path would in place instead of being transmitted c_i times. Such saving is then averaged over the total measurement time considered in this study, i.e. 30 minutes.

Figure 7 shows the results obtained for the percentage of traffic savings over time frame t. From this figure, we can observe that the percentage of savings provided by a multicast scheme in the case of Web/HTTP traffic application is relatively low even considering high time frame t such as 3 seconds (7% of savings). Note that increasing the time frame to more than 3 seconds is expected to yield higher savings. We emphasize though that this study has been realized on an NREN network and not a commercial network.

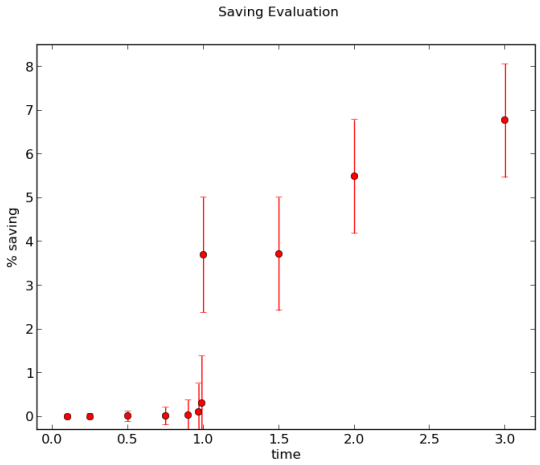


Figure 7. Percentage of traffic savings as a function of time frame t

4.3 Peer-to-peer traffic analysis

In this section, we analyze the P2P traffic to reveal the locality properties of receivers in live P2P streaming systems. The choice of the streaming content was driven by the subjectively perceived importance of the ongoing events at the time the experiment took place. With this in mind, we captured the content streamed by several SOPCast channels during a UEFA Champions League semifinal match. In spite of the fact that interest for such football matches is highest in Europe, the teams involved are both highly appreciated worldwide and amount players spanning many nationalities.

For the capturing process, we used 7 vantage pointsspanning a total of 6 countries: 2 vantage points in USA (California and Virginia), 3 in Europe (Ireland, Barcelona and Cluj (Romania)) and 2 in Asia (Singapore and Tokyo). The objective is to create

an infrastructure capable of performing a world-wide distributed passive capture of large P2P live content streaming overlays. All machines involved ran Ubuntu Linux and have a 100Mbps Ethernet card; four different channels (at different bit-rates) have been analyzed.

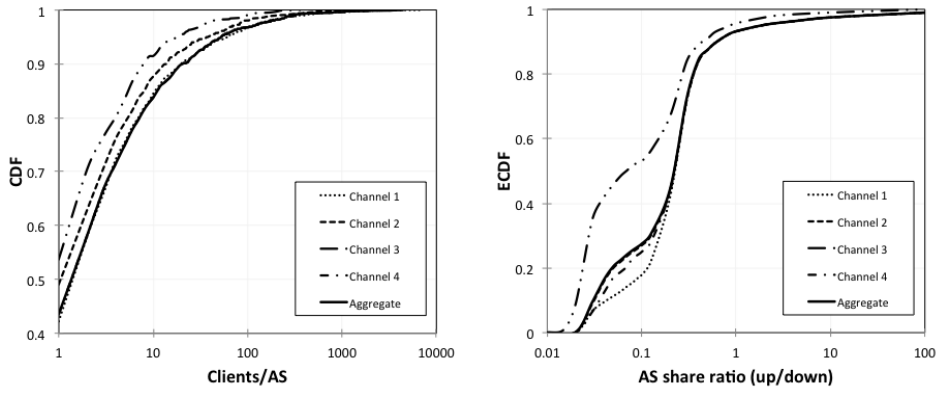
Some of the statistical properties of the traces captured are presented in Table 1. Among them, we count the peering IPs encountered in each trace, which may be used as an estimate for the number of connected end-hosts. However, because we did not identify hosts behind NAT devices, this value should be held as a lower bound estimate. From the peer IPs, the number of Autonomous Systems (AS) exchanging traffic with our nodes is inferred. We define a similarity metric in order to evaluate the breadth of the peer and AS population that we measured. For a vantage point in a channel, the metric was defined as the ratio of IPs/ASNs that overlap with those encountered in traces from other vantage points. The high values measured for IP similarity indicate that in each channel our vantage points exchanged traffic with a large fraction of the peer population, leading to an accurate aggregate view of the whole overlay. Furthermore, AS similarity values suggest that we have a precise estimate of the AS exchanging traffic. Overall, we can infer that streaming channels generally have non-overlapping clients (as expected); however, their clients belong to ASthat have a large overlap.

Table 1. Trace properties

Ch1@850kbps	Download (GBytes)	Number of IPs	Number of AS	Up (%)	Down (%)	Similarity IP (%)	Similarity AS (%)
California	1.45	19250	1469	76	24	93	98
Cluj	1.50	32229	1980	90	10	92	96
Ireland	0.99	13522	1294	66	34	92	98
Barcelona	1.31	34320	1940	91	9	78	94
Singapore	1.39	37164	2039	89	11	89	96
Tokyo	1.18	37822	2028	95	5	91	96
Virginia	1.33	21864	1745	88	12	92	96
Total	9.63	64586	2839	89	11	12	68

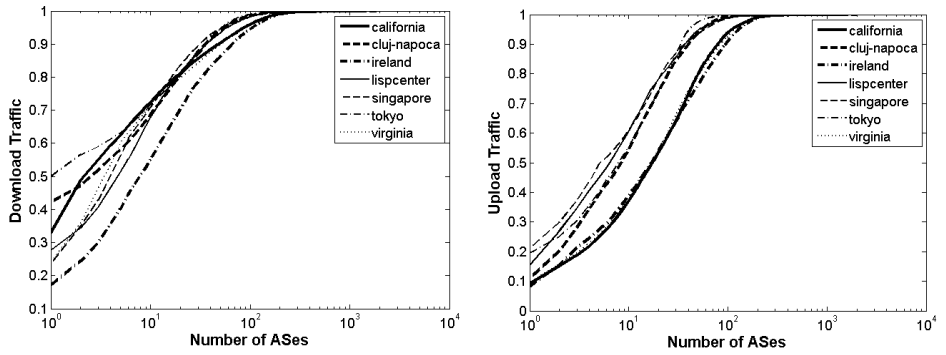
The distribution of clients in AS is depicted in Figure 8a. It can be seen that the plots have a similar shape and the differences between them are only due to the inter-channel client variations. As we have seen in Table 1, channels tend to have non-overlapping client populations. Therefore the reasons behind the similarity of the curves have to do with a subtler phenomenon probably related to user behavior and localized user interest.

Within P2P systems, it is the responsibility of the peers to replicate content to other members. In Figure 8b we study the amount of traffic exchanged by our nodes with their AS peers. To evaluate their level of collaboration, we define and compute a sharing ratio for each overlay member. Specifically, for each peer, the sharing ratio is computed by dividing its volume of uploaded traffic by the download one.



a) b)

Figure 8.a) Clients per AS, and b) Distribution of clients in AS



a) b)

Figure 9. Amount of a) download and b) upload traffic exchanged by VP nodes with their peers

Finally, Figure 9 illustrates the degree of collaboration between peers for both download and upload traffic. In absolute terms, the plots show that our nodes obtain between 20 and 50% of the content from peers in the same AS reaching 100% with around 100 AS. On the other hand, our nodes act as seeds for few peers in the same AS (between 10 and 20%) while all peers are in around 100 AS.These results indicate that the majority of the peers are localized outside the AS containing all vantage points. In turn, this means that a multicast scheme could potentially provide an advantage support for inter-domain P2P traffic.

5 Conclusion

Aiming at applying the iterative research process at the inception of the FIRE initiative, we present in this paper its application to the development of a dynamic multicast routing scheme that would be able to overcome the known architectural and scaling limits of current protocol independent multicast routing (such as PIM) but also provide a suitable alternative to existing compact multicast routing schemeswhich exhibit a considerable problem: the sparse cover underlying the execution of the Abraham scheme is constructed off-line and requires global knowledge of the network topology to properly operate.

In terms of performance (Section 3.1), the proposed GCMR scheme shows substantial gain in terms of the number of RT entries compared to the Steiner-Tree (ST) heuristic (minimum factor of 3,21 for sets of 4000 leaf nodes, i.e., 12,5% of the topology size) and the memory space required to store them. The stretch deterioration compared to the ST algorithms ranges between 8% and 3% (for multicast group size of 500 to 4000, respectively); thus, decreasing with increasing group sizes. The proposed two-phase search process -local search first covering the leaf's node vicinity, and if unsuccessful, a global search over the remaining topology -combined with the vicinity ball construction at the source node- enables to keep the communication cost of the GCMR algorithm within reasonable bounds compared to the reference Shortest Path Tree (SPT) scheme and sub-linearly proportional to the size of the leaf node set.
Comparison with the Abraham scheme (Section 3.2), the GCMR scheme provides abetter tradeoff between the memoryspace required to store the RT entries and the stretch factor increase of the produced multicast routing paths.

Having compared performance, the critical question becomes how to take advantage of these properties; indeed, the spatio-temporal distribution of traffic must exhibit locality in order to taking benefit of multicast routing. Initial results (obtained from Catalan NREN access point to the Internet) show a potential gain of less than 10% for Web/HTTP traffic. For “multimedia” traffic, this second level analysis still needs to be conducted. Moreover, the tool will be packaged in order to provide the mean for other NREN to perform similar traffic analysis studies. Ultimately, it would be interesting to initiate traffic captures at different NRENs during same time periods and perform traffic analysis across multiple NRENs.

Acknowledgements

This research work is conducted by the EULER Project (Grant No.258307)part of the Future Internet Research and Experimentation (FIRE) objective of the Seventh Framework Programme (FP7) funded by the European Commission (EC).

References

1. B.Fenner et.al., *Protocol Independent Multicast - Sparse Mode (PIM-SM)*, Internet Engineering Task Force (IETF), RFC 4601, Aug.2006.
2. T.Ballardie, P.Francis and J.Crowcroft, *Core Based Trees (CBT): An Architecture for Scalable Multicast Routing*, Proceedings of ACM Sigcomm, pp.85-95, 1995.
3. H.Holbrook and B.Cain, *Source-Specific Multicast for IP*, Internet Engineering Task Force (IETF), RFC 4607, Aug.2006.
4. C.Diot et al., *Deployment Issues for the IP Multicast Service and Architecture*, IEEE Network, vol.4, no.1, pp.78-88, Jan/Feb.2000.
5. EULER FP7 Project, *Performance objectives, evaluation criteria and metrics*, Technical report. Available at: https://www-sop.inria.fr/mascotte/EULER/wiki/pmwiki.php/Main/Deliverables
6. P.Pedroso, D.Papadimitriou, D.Careglio, *Dynamic compact multicast routing on power law graphs*, 54th IEEE Globecom, Houston (TX), USA, Dec.2011.
7. I.Abraham, D.Malkhi, and D.Ratajczak, *Compact multicast routing*, Proceedings of 23rd International Symposium DISC'09, Elche, Spain, pp.364–378, Sep.2009.
8. T.Bates et al., *Multiprotocol Extensions for BGP-4*, Internet Engineering Task Force (IETF), RFC 4760, Jan.2007.
9. E.Rosen and R.Aggarwal (Ed's), *Multicast in MPLS/BGP IP VPNs*, Internet Engineering Task Force (IETF), RFC 6513, Feb.2012.
10. T.Bu, and D.Towsley, *On distinguishing between Internet power law topology generators*, Proc. 21th Annual IEEE International Conference on Computer Communications (INFOCOM 2002), vol.2, pp.638-647, 2002.
11. Sage's Graph Library. Available at http://www.sagemath.org/
12. V.Carela-Español, P.Barlet-Ros, A.Cabellos-Aparicio, and J.S.-Pareta, *Analysis of the impact of sampling on NetFlow traffic classification*, Computer Networks, 55(5), Apr. 2011.
13. AnellaCientifica, http://www.cesca.cat/en/communications/anella-cientifica
14. RedIris, http://www.rediris.es
15. Cisco NetFlow. http://www.cisco.com/warp/public/732/Tech/netflow.
16. J.Quinlan, *C4.5: programs for machine learning*, Morgan Kaufmann, 1993.
17. P.Phaal, M.Lavine, *sFlow Version 5*, sFlow.org. July 2004.
18. Sampled NetFlow, http://www.cisco.com/en/US/docs/ios/12_0s/feature/guide/12s_sanf.html.